Solutions. Experts. Insights.

SEI TECHNOLOGIES FORUM

Software Engineering Institute | Carnegie Mellon

A Brief Survey of the Team Software Process℠ (TSP℠)

James McHale
Team Software Process (TSP) Initiative.

McHale is a senior member of the technical staff at the SEI. Prior to joining the SEI in 1999, he spent more than 20 years in industry as a software engineer, system designer, project leader, and development manager working on control systems for diverse applications such as steel mills, power plants, robotics, and transportation.

.

© 2011 Carnegie Mellon University

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
24 OCT 2011		**00-00-2011 to 00-00-2011**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
A Brief Survey of the Team Software ProcessSM (TSPSM)	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Carnegie Mellon University ,Software Engineering Institute,Pittsburgh,PA,15213**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**31**	

Topics

What is TSP?

Does TSP Work?

How Does TSP Work?

Is TSP Agile?

Does TSP Displace Other Practices?

© 2011 Carnegie Mellon University

Topics

What is TSP?

Does TSP Work?

How Does TSP Work?

Is TSP Agile?

Does TSP Displace Other Practices?

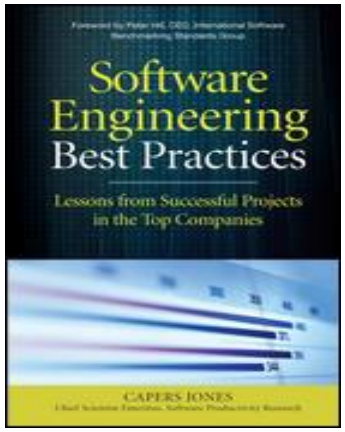
© 2011 Carnegie Mellon University

TSP – A World-Class Development Practice

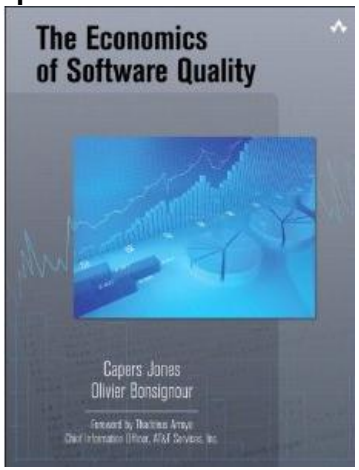
Rank	Small Systems <1K FP	Medium Systems 1K FP to 100K FP	Large Systems >100K FP
1	Agile	TSP	TSP
2	Agile	CMMI L4, L5	CMMI L4, L5
3	Waterfall	CMMI L3	CMMI L3
4	CMMI L2	RUP	Hybrid

Development practices by size of application in function points (FP) [1] [2]

(1FP ≈ 30 to 50 SLOC)



[1] Software Engineering Best Practices, by Capers Jones, 2010.



[2] The Economics of Software Quality, by Capers Jones, 2011.

© 2011 Carnegie Mellon University

Team Software Process

The Team Software Process (TSP) is a process framework that was *initially* designed for software teams.

It's purpose is to help teams achieve their best performance by showing them how to

- accurately estimate and plan their work
- negotiate their commitments with management
- manage and track projects to a successful conclusion
- manage quality to produce better products in less time

© 2011 Carnegie Mellon University

TSP – No Longer "Just for Software"

Even from it's earliest pilot projects, TSP was used by integrated project teams that included various domain experts.

- electronic and mechanical engineers
- system engineers
- test engineers
- business analysts
- technical writers
- game and graphic designers

In recent years, TSP has been adapted for use by non-software knowledge-working teams.

- process groups
- nuclear engineers
- systems & test engineers
- IT services

© 2011 Carnegie Mellon University

Organizations Using TSP

Advanced Information Services, Inc.
Beckman Coulter
Bursatec
Cadence
Centro De Investigacion En Matamaticas
Chinasoft International, Inc.
COmputing TechnologieS, Inc.
Davis Systems
DEK International GmbH
Delaware Software, S.A. de C.V.
Delivery Excellence
Grupo Empresarial Eisei, S.A. de C.V.
Herbert Consulting
Hitachi Software Engineering Co., Ltd.
Idea Entity Corp.
InnerWorkings, Inc.
Instituto Tecnologico y de Estudios Superiores de Monterrey

Johannesburg Centre for Software Engineering
Kernel Technologies Group, S.A. de CV
Knowldege Partner QR Pvt. Ltd.
Kyushu Institute of Technology
L. G. Electronics
LogiCare
Motiva, LLC
National Aeronautics & Space Administration
Nedbank
Next Process Institute Ltd.
Praxis High Integrity Systems
Process & Project Health Services
Procesix
PS&J Consulting - Software Six Sigma
QuarkSoft
Sandia National Laboratories
Science Applications International Corporation
Seontis

Siemens AG
SILAC Ingenieria de Software S.A. de C.V.
SKIZCorp Technology
Software Engineering Competence Center (SECC)
Software Park Thailand
STPP, Inc.
TOWA INTEGRADADORA S.A. de C.V.
TRX
Universidad Autonoma De Zacatecas
Universidad de Monterrey
Universidad Regiomotana A.C.
University of Aizu
U.S. Air Force (CRSIP/STSC)
U.S. Census Bureau
U.S. Navy Air Systems Command (NAVAIR)
U.S. Naval Oceanographic Office (NAVO)
Urban Science

© 2011 Carnegie Mellon University

Topics

What is TSP?

Does TSP Work?

How Does TSP Work?

Is TSP Agile?

Does TSP Displace Other Practices?

© 2011 Carnegie Mellon University

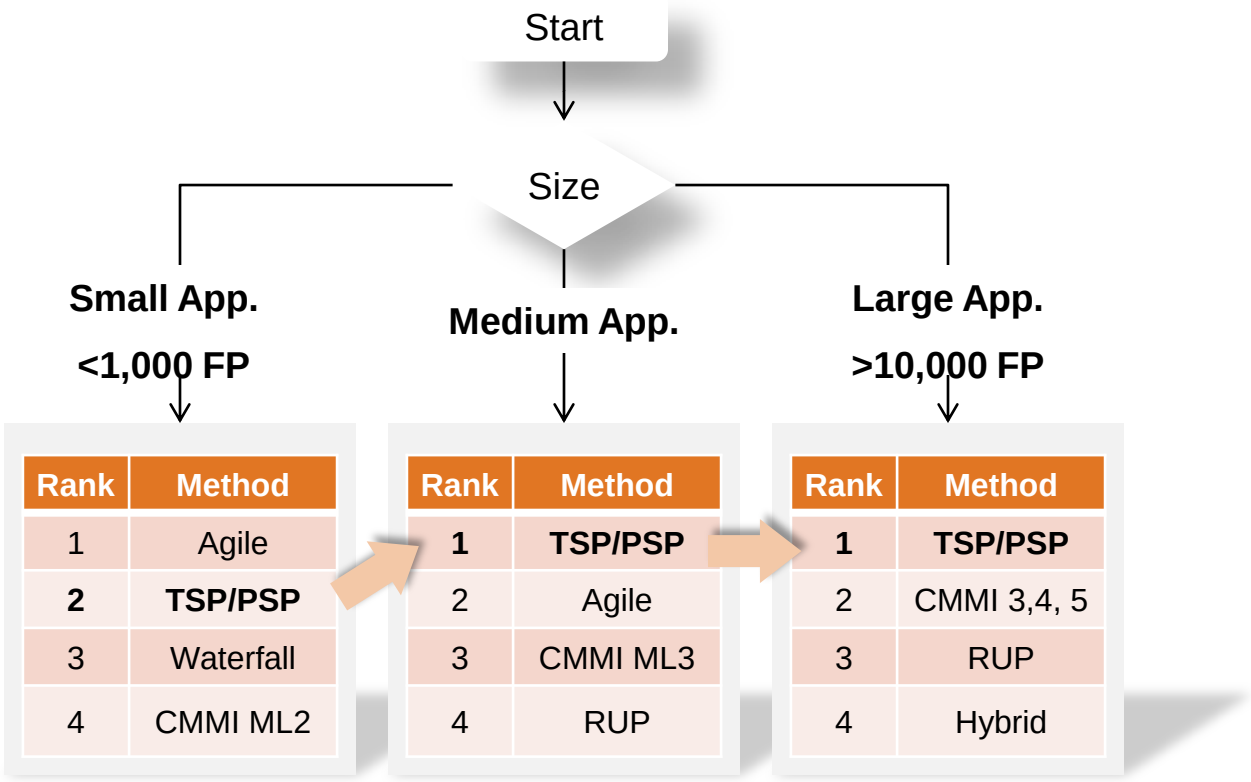
Project Settings Where TSP Has Been Applied

Project Team Characteristic	Experience
Team Size	From two to twenty team members per team and multi-team projects up to 100+ developers
Team Location	Co-located, geographically distributed, and even multi-team settings with more than one organization participating.
Work Domain	Software engineering, hardware teams, systems engineering, nuclear engineering, and IT services.
Multi-discipline	Integrated teams consisting of many disciplines such as software, hardware, test and quality assurance, business analysts, architects, game developers, artists, documentation specialists, and other engineering disciplines.
Project Duration	Very short projects of a few weeks duration up to projects lasting a few years.
Development Strategies	Mostly incremental or iterative development with periodic internal and external releases. Cycle or sprint length ranging from a few weeks to a few months.
Project Mode	New systems, enhancements to existing systems, including very large legacy systems, pure maintenance, prototyping, system feasibility, system proposal teams, and projects or groups providing on-going and request-based services.

© 2011 Carnegie Mellon University

Scalability

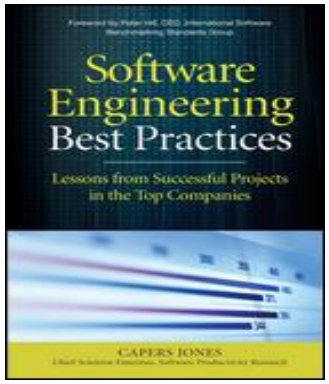
Capers Jones ranked TSP as the best software engineering practice for medium and large applications and a close second for small applications.



Rank	Method
1	Agile
2	**TSP/PSP**
3	Waterfall
4	CMMI ML2

Rank	Method
1	**TSP/PSP**
2	Agile
3	CMMI ML3
4	RUP

Rank	Method
1	**TSP/PSP**
2	CMMI 3,4, 5
3	RUP
4	Hybrid



Software Engineering Best Practices,

C. Jones, 2010

© 2011 Carnegie Mellon University

Reported Benefits

Microsoft IT

- Mean schedule error reduced from 10% to 1%
- Substantially reduced post-code complete defects

Intuit

- QuickBooks post-release defects reduced by 60%
- Functionality delivered increased by 30%

Adobe

- Four times less rework on TSP teams
- Adoption of TSP is a great way for teams to improve both quality and productivity

Oracle

- 7 fold improvement in quality
- System Test cycle effort reduced by 90%

Vicarious Visions

- *"TSP helped us meet schedule while addressing a zero-defect quality objective"*

© 2011 Carnegie Mellon University

Topics

What is TSP?

Does TSP Work?

How Does TSP Work?

Is TSP Agile?

Does TSP Displace Other Practices?

© 2011 Carnegie Mellon University

Knowledge Work

"The key rule in managing knowledge work is this: managers can't manage it, the workers must manage themselves."

Software development is knowledge work.

To manage software work, developers must

- be motivated
- make accurate plans
- negotiate commitments
- track their plans
- manage quality

How is this accomplished?

Watts Humphrey, creator of TSP

© 2011 Carnegie Mellon University

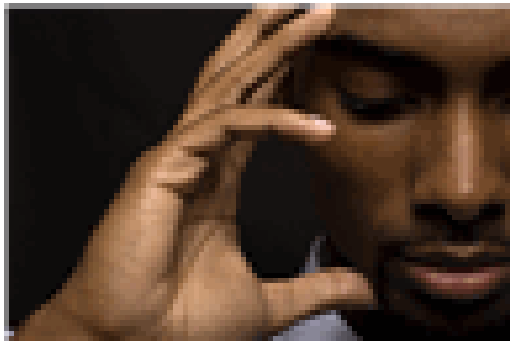
Personal Software Process (PSP)

PSP Course	
PSP Level	**Focus**
PSP 0	Basic method Metrics
PSP 1	Planning Estimating Tracking
PSP 2	Design Quality management

PSP skills are the foundations of TSP.

PSP training introduces team skills using a progressive model.

Developers try new practices and learn from their results.



© 2011 Carnegie Mellon University

The TSP/PSP Measurement Framework

Schedule

Effort

Size

Quality

Source: CMU/SEI-92-TR-019

Four direct measures apply to all processes and products.

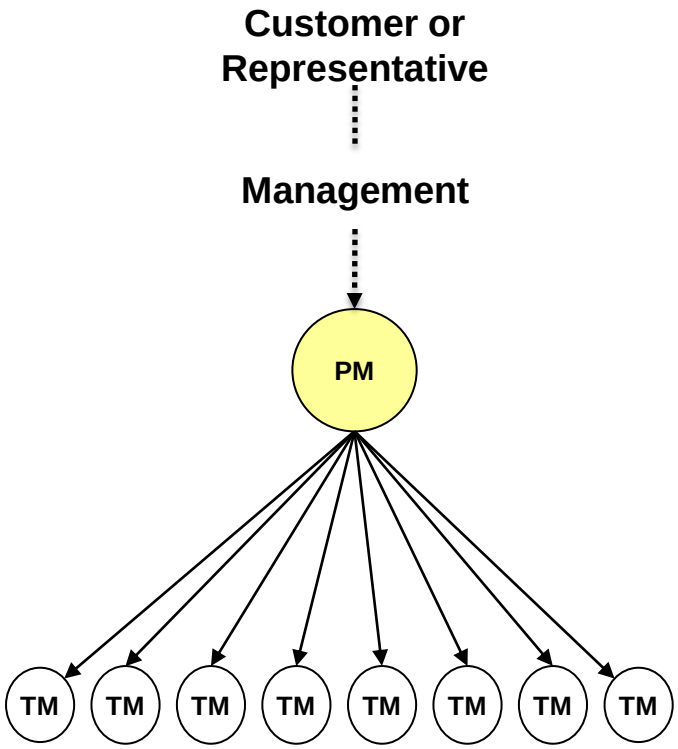
- estimates are made during planning
- measures are recorded by developers while working

The data are used to track project status and to analyze and improve performance.

Benefit – direct measures, integrated into a measurement framework, provide flexibility.

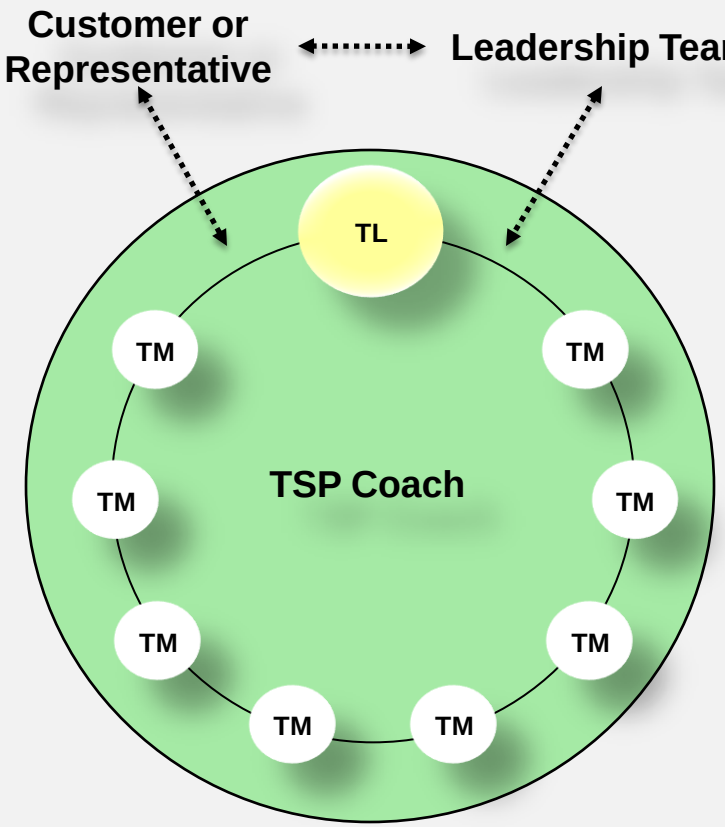
© 2011 Carnegie Mellon University

Team Management Styles



Traditional team

The leader plans, directs, and tracks the work.



TSP Self-directed team

The team plans, directs, and tracks their work.

© 2011 Carnegie Mellon University

The TSP Coaching Role

The coach

- trains and facilitates the adoption of TSP
- works with the team leader to build the team
- observer that guides the team

Team Leader vs. Coach

The team leader's job is to use the team to build the product.

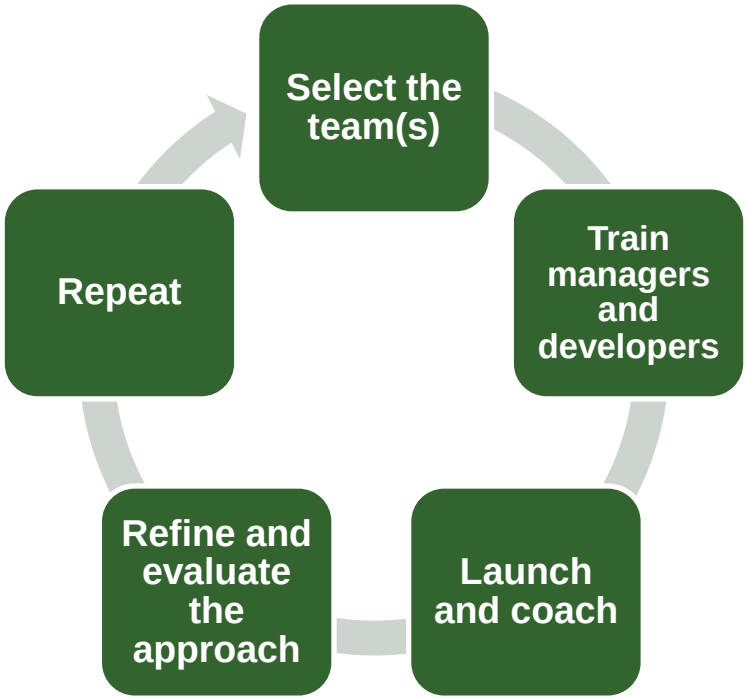
The coach's job is to use the project to build the team.

© 2011 Carnegie Mellon University

Rapid Deployment Strategy

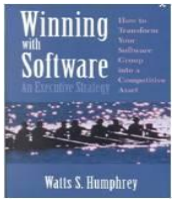
TSP is implemented project-by-project.

- Select two or three teams.
- Train top-down, starting with senior managers, then project managers, then team members.
- When the managers and team are trained, conduct a TSP Launch to kick-off each project.
- Evaluate and fine tune the approach.
- Repeat this cycle increasing scope at a sustainable pace.



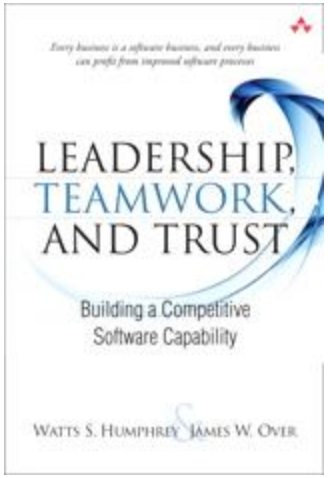
© 2011 Carnegie Mellon University

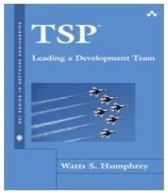
TSP Training Courses



TSP Executive Strategy Seminar

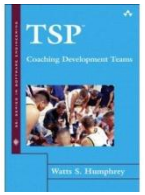
- Building a “winning” organization
- Managing with facts and data
- One-day course





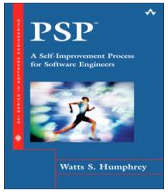
Leading a Development Team

- Building self-directed teams
- Motivating and leading self-directed teams
- Three-day course



Coaching Development Teams

- Launching Teams
- Coaching teams
- Five-day course



PSP for Software Developers

- Using a defined and measured personal process
- Planning, tracking, design, quality management
- Five-day course

© 2011 Carnegie Mellon University

Topics

What is TSP?

Does TSP Work?

How Does TSP Work?

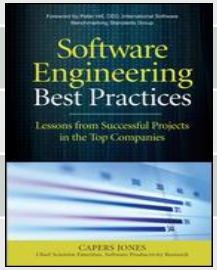
Is TSP Agile?

Does TSP Displace Other Practices?

© 2011 Carnegie Mellon University

TSP *is* Agile – Plus…

Features TSP and Agile Methodologies Support	
Agile	**TSP**
Team organization	Team organization
Project management – planning and estimating	Project management – planning and estimating
Change control	Change control
Requirements	Requirements
Design	Design
Code development	Code development
Configuration control	Configuration control
Testing	Testing
	Specialization of team members
	Project management – tracking and control
	Reusability
	Quality assurance
	Inspections
	Static analysis
	Security
	Documentation and training



Software Engineering Best Practices, C. Jones, 2010

© 2011 Carnegie Mellon University

Quality Management is Critical

Jeff Sutherland, inventor of SCRUM, recently wrote online of the need for “intolerance of defects” in managing product backlog. ***But how?***

From data on over 40 TSP teams, Intuit found that

- post code-complete effort was 8% instead of 33% of the project
- for TSP projects, standard test times were cut from 4 months to 1 week
- most TSP teams included some agile practices in their team processes

Testing time was reduced from four months to one month.

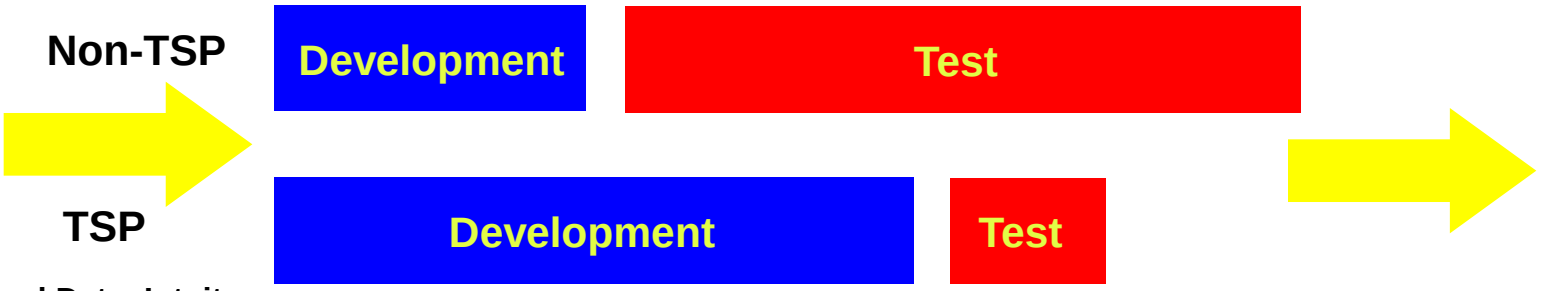


Diagram and Data: Intuit

© 2011 Carnegie Mellon University

Topics

What is TSP?

Does TSP Work?

How Does TSP Work?

Is TSP Agile?

Does TSP Displace Other Practices?

© 2011 Carnegie Mellon University

CMMI and TSP

CMMI is a model that describes many of the best practices for development.

- about “what” not “how-to”
- an improvement roadmap
- a capability benchmark

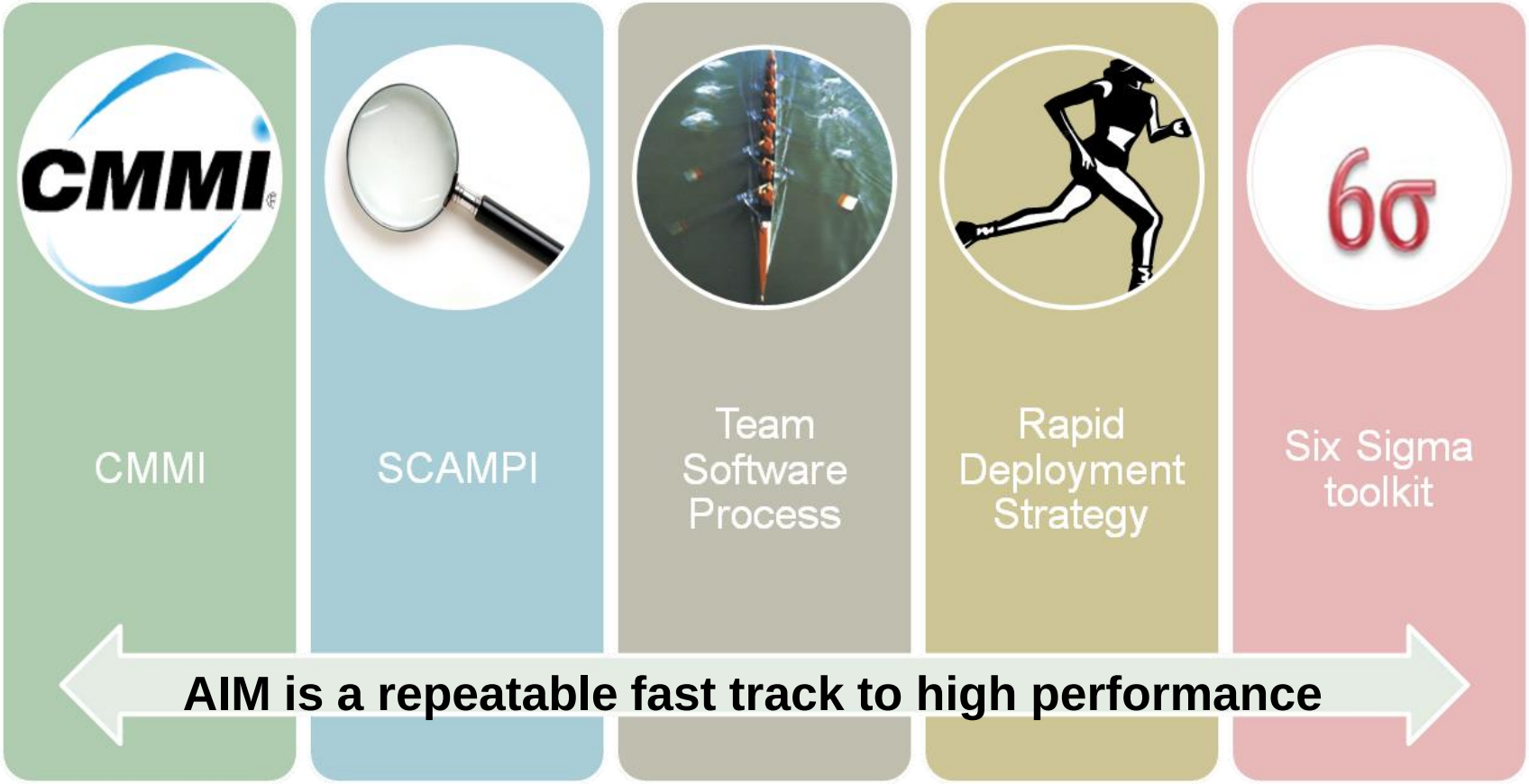
TSP is a process that integrates many CMMI best practices.

- about “how-to” not “what”
- an improvement tool
- a performance benchmark

© 2011 Carnegie Mellon University

Accelerated Improvement Method (AIM)

Integrates and Leverages Effective Improvement Technologies



© 2011 Carnegie Mellon University

A Challenging Project

Background:

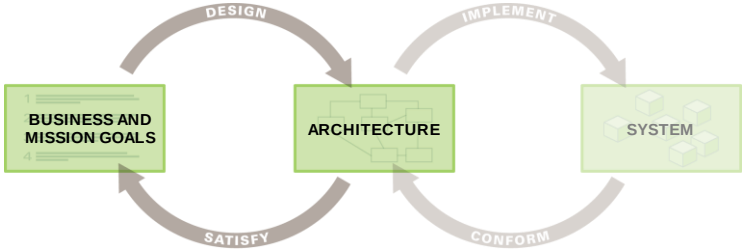
- Bolsa Mexicana de Valores (BMV) operates the Mexican financial markets under license from the federal government.
- Bursatec is the technology arm of the BMV.
- BMV desired a new trading engine to replace the existing stock market engine and integrate the options and futures markets.
- The BMV performed a build vs. buy analysis, and decided to replace their three existing trading engines with one in-house developed system.

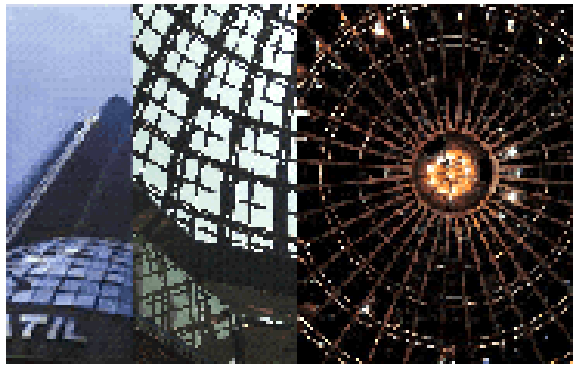
Approach:

- Build the right system (SEI Architecture)
- Build the right way (SEI TSP)

© 2011 Carnegie Mellon University

Challenging Project's Results





A key TSP principle:

Doing it the right way is, in the long run, always faster and cheaper.

- On time (21 months of development including 5 major milestones)
- Slightly under budget (14-17 developers including architects, no use of contingency)
- Few defects (tests ongoing but < 50 actual defects to date in 200,000+ LOC)
- ALL key quality attributes achieved (main latency requirement exceeded by factor of 10); extra functionality implemented
- NO SURPRISES! NO FIRE-DRILLS! (project was transparent both technically and from the Project Office view)
- Very low turnover (1 from project, 1 after leaving the project)

© 2011 Carnegie Mellon University

Summary

TSP – the Team Software Process

- is a robust, agile project and quality management framework
- is for project teams up to 100 or more
- has world-class, quantified, published results
- has proven, repeatable methods for training, introduction, and coaching

TSP projects successfully integrate other best practices, including

- most agile methods, including many elements of SCRUM and XP
- RUP and most other lifecycle methodologies
- Six Sigma
- CMMI at all maturity levels
- SEI architecture and secure coding practices

© 2011 Carnegie Mellon University

Contact Information

James McHale
Sr. Member of the Technical Staff
TSP Initiative
Telephone: +1 412-268-3948
Email: jdm@sei.cmu.edu

U.S. Mail
Software Engineering Institute
Customer Relations
4500 Fifth Avenue
Pittsburgh, PA 15213-2612
USA

Web
www.sei.cmu.edu/TSP
www.sei.cmu.edu/TSPSymposium

Customer Relations
Email: info@sei.cmu.edu
Telephone: +1 412-268-5800
SEI Phone: +1 412-268-5800
SEI Fax: +1 412-268-6257

© 2011 Carnegie Mellon University

NO WARRANTY

THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN “AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Use of any trademarks in this presentation is not intended in any way to infringe on the rights of the trademark holder.

This Presentation may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

SEPG 2012 NORTH AMERICA | Reaching New Levels of Excellence

Software Engineering Institute | Carnegie Mellon

© 2011 Carnegie Mellon University